EXHIBIT N

JENAM TECH LLC'S INFRINGEMENT ANALYSIS

U.S. Patent No. 9,923,996 – Google LLC
Claim 1

Jenam Tech LLC ("Jenam") provides evidence of infringement of Claim 1 of U.S. Patent No. 9,923,996 (hereinafter "the '996 patent") by Google LLC ("Google"). In support thereof, Jenam provides the following claim charts.

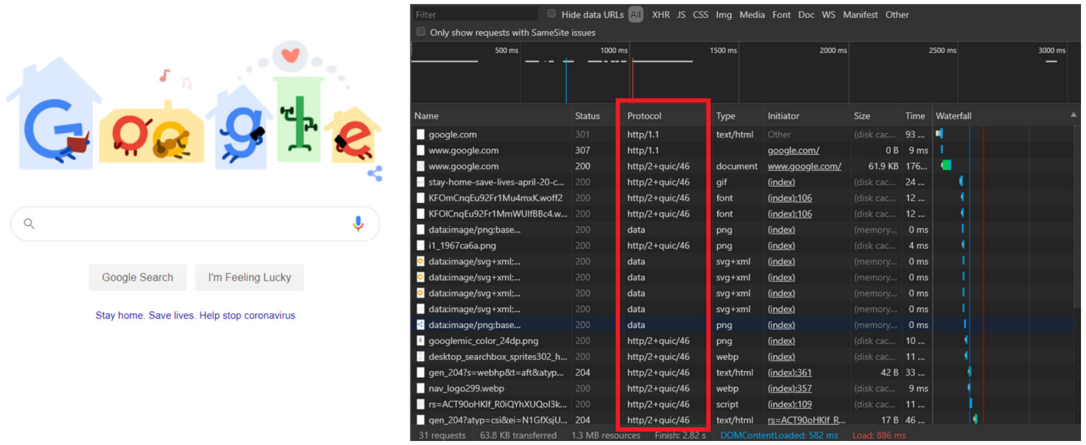
"Accused Instrumentalities" as used herein refers to at least one or more websites or web addresses including, but not limited to www.google.com, stored and/or hosted on one or more servers owned or under the control of Google. These claim charts demonstrate Google's infringement, and provide notice of such infringement, by comparing each element of the asserted claims to corresponding components, aspects, and/or features of the Accused Instrumentalities. These claim charts are not intended to constitute an expert report on infringement. These claim charts include information provided by way of example, and not by way of limitation.

The analysis set forth below is based only upon information from publicly available resources regarding the Infringing Instrumentalities, as Google has not yet provided any non-public information. An analysis of Google's (or other third parties') technical documentation and/or software source code may assist in fully identify all infringing features and functionality. Accordingly, Jenam reserves the right to supplement this infringement analysis once such information is made available to Jenam. Furthermore, Jenam reserves the right to revise this infringement analysis, as appropriate, upon issuance of a court order construing any terms recited in the asserted claims.

Unless otherwise noted, Jenam contends that Google directly infringes the '996 patent in violation of 35 U.S.C. § 271(a) by selling, offering to sell, making, using, and/or importing the Infringing Instrumentalities. The following exemplary analysis demonstrates that infringement. Unless otherwise noted, Jenam further contends that the evidence below supports a finding of indirect infringement under 35 U.S.C. §§ 271(b) and/or (c), in conjunction with other evidence of liability under one or more of those subsections. Google makes, uses, sells, imports, or offers for sale in the United States, or has made, used, sold, imported, or offered for sale in the past, without authority, or induces others to make, use, sell, import, or offer for sale in the United States, or has induced others to make, use, sell, import, or offer for sale in the past, without authority products, equipment, or services that infringe Claim 1 of the '996 patent, including without limitation, the Accused Instrumentalities.

Unless otherwise noted, Jenam believes and contends that each element of each claim asserted herein is literally met through Google's provision of the Infringing Instrumentalities. However, to the extent that Google attempts to allege that any asserted claim element is not literally met, Jenam believes and contends that such elements are met under the doctrine of equivalents. More specifically, in its investigation and analysis of the Infringing Instrumentalities, Jenam did not identify any substantial differences between the elements of the patent claims and the corresponding features of the Infringing Instrumentalities, as set forth herein. In each instance, the identified feature of the Infringing Instrumentalities performs at least substantially the same function in substantially the same way to achieve substantially the same result as the corresponding claim element.

To the extent the chart of an asserted claim relies on evidence about certain specifically-identified Accused Instrumentalities, Jenam asserts that, on information and belief, any similarly-functioning instrumentalities also infringes the charted claim. Jenam reserves the right to amend this infringement analysis based on other products made, used, sold, imported, or offered for sale by Google. Jenam also reserves the right to amend this infringement analysis by citing other claims of the '996 patent, not listed in the claim chart, that are infringed by the Accused Instrumentalities. Jenam further reserves the right to amend this infringement analysis by adding, subtracting, or otherwise modifying content in the "Accused Instrumentalities" column of each chart.

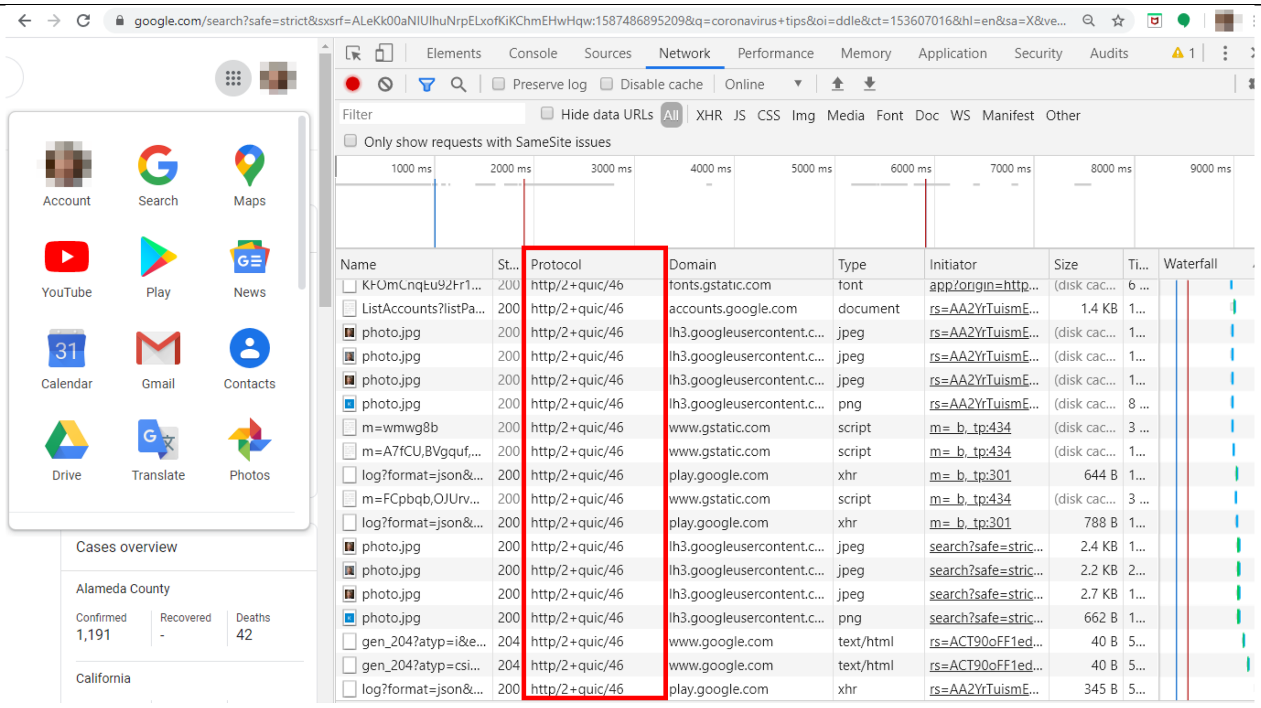
Claim 1	**Accused Instrumentalities**
An apparatus comprising: a non-transitory memory storing a network application; and one or more processors in communication with the non-transitory memory, wherein the one or more processors execute the network application such that the network application is configured to operate in accordance with a first protocol including a transmission control protocol (TCP), the apparatus, when operating in accordance with the first protocol to establish a TCP connection, configured to:	Google uses an apparatus (e.g., one or more servers, etc.) including a non-transitory memory storing a network application (e.g., server software, etc.), and one or more processors in communication with the non-transitory memory, wherein the one or more processors execute the network application such that the network application is configured to operate in accordance with a first protocol including a transmission control protocol (TCP). See excerpt(s) below, for example (emphasis added, if any): __Note__: Below is a web page of Google (https://www.google.com/). 

Name	Status	Protocol	Type	Initiator	Size	Time	Waterfall
google.com	301	http/1.1	text/html	Other	(disk cac...	93 ...	
www.google.com	307	http/1.1		google.com/	0 B	9 ms	
www.google.com	200	http/2+quic/46	document	www.google.com/	61.9 KB	176...	
stay-home-save-lives-april-20-c...	200	http/2+quic/46	gif	(index)	(disk cac...	24 ...	
KFOmCnqEu92Fr1Mu4mxK.woff2	200	http/2+quic/46	font	(index):106	(disk cac...	12 ...	
KFOlCnqEu92Fr1MmWUlfBBc4.w...	200	http/2+quic/46	font	(index):106	(disk cac...	12 ...	
data:image/png;base...	200	data	png	(index)	(memory...	0 ms	
i1_1967ca6a.png	200	http/2+quic/46	png	(index)	(disk cac...	4 ms	
data:image/svg+xml;...	200	data	svg+xml	(index)	(memory...	0 ms	
data:image/svg+xml;...	200	data	svg+xml	(index)	(memory...	0 ms	
data:image/svg+xml;...	200	data	svg+xml	(index)	(memory...	0 ms	
data:image/svg+xml;...	200	data	svg+xml	(index)	(memory...	0 ms	
data:image/png;base...	200	data	png	(index)	(memory...	0 ms	
googlemic_color_24dp.png	200	http/2+quic/46	png	(index)	(disk cac...	10 ...	
desktop_searchbox_sprites302_h...	200	http/2+quic/46	webp	(index)	(disk cac...	11 ...	
gen_204?s=webhp&t=aft&atyp...	204	http/2+quic/46	text/html	(index):361	42 B	33 ...	
nav_logo299.webp	200	http/2+quic/46	webp	(index):357	(disk cac...	9 ms	
rs=ACT90oHKIf_R0iQYhXUQol3k...	200	http/2+quic/46	script	(index):109	(disk cac...	11 ...	
gen_204?atyp=csi&ei=N1GfXsjU...	204	http/2+quic/46	text/html	rs=ACT90oHKIf_R...	17 B	46 ...	

31 requests | 63.8 KB transferred | 1.3 MB resources | Finish: 2.82 s | DOMContentLoaded: 582 ms | Load: 886 ms

communicate a segment including at least one first synchronize bit; communicate a first acknowledgement of the segment, and at least one second synchronize bit; and	Google uses the apparatus (e.g., the one or more servers, etc.) that, when operating in accordance with the first protocol to establish the TCP connection, is configured for a) communicating a segment including at least one first synchronize bit; b) communicating a first acknowledgement of the segment, and at least one second synchronize bit; and c) communicating a second acknowledgement. See excerpt(s) below, for example (emphasis added, if any): __Note__: As set forth below, a) is met by 1); b) is met by 2)/3), and c) is met by 4).

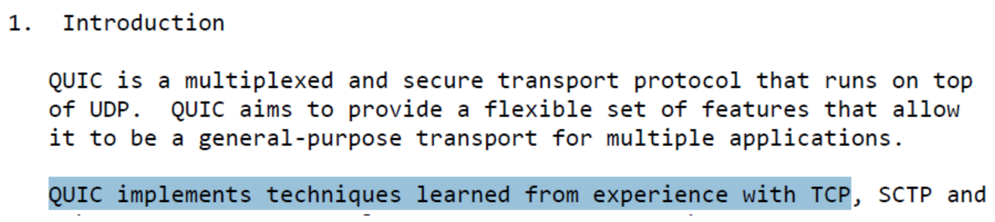
communicate a second acknowledgement;	“The synchronization requires each side to send its own initial sequence number and to receive a confirmation of it in acknowledgment from the other side. Each side must also receive the other side's initial sequence number and send a confirming acknowledgment. 1) A --> B SYN my sequence number is X 2) A <-- B ACK your sequence number is X 3) A <-- B SYN my sequence number is Y 4) A --> B ACK your sequence number is Y Because steps 2 and 3 can be combined in a single message this is called the three way (or three message) handshake. A three way handshake is necessary because sequence numbers are not tied to a global clock in the network, and TCPs may have different mechanisms for picking the ISN's. The receiver of the first SYN has no way of knowing whether the segment was an old delayed one or not, unless it remembers the last sequence number used on the connection (which is not always possible), and so it must ask the sender to verify this SYN. The three way handshake and the advantages of a clock-driven scheme are discussed in [3].” “Request for Comments” (RFC) document RFC 793 (September 1981) https://tools.ietf.org/html/rfc793
wherein the network application is further configured to operate in accordance with a second protocol that is separate from the TCP, the apparatus, when operating in accordance with the second protocol to establish a second protocol connection, configured to: receive, by a second node from a first node, a packet;	Google uses the apparatus (e.g., the one or more servers, etc.) including the non-transitory memory storing the network application (e.g., server software, etc.) that is further configured to operate in accordance with a second protocol (e.g., QUIC, etc.) that is separate from the TCP. The apparatus, when operating in accordance with the second protocol to establish a second protocol connection (e.g., QUIC connection), is configured to: receive, by a second node (e.g., one of a QUIC-compliant server or client) from a first node (e.g., the other one of the QUIC-compliant server or client), a packet (e.g., QUIC negotiation packet). See excerpt(s) below, for example (emphasis added, if any): **Note**: Below is a web page of Google (https://www.google.com/).



google.com/search?safe=strict&sxsrf=ALeKk00aNIUlhuNrpELxofKiKChmEHwHqw:1587486895209&q=coronavirus+tips&oi=ddle&ct=153607016&hl=en&sa=X&ve...

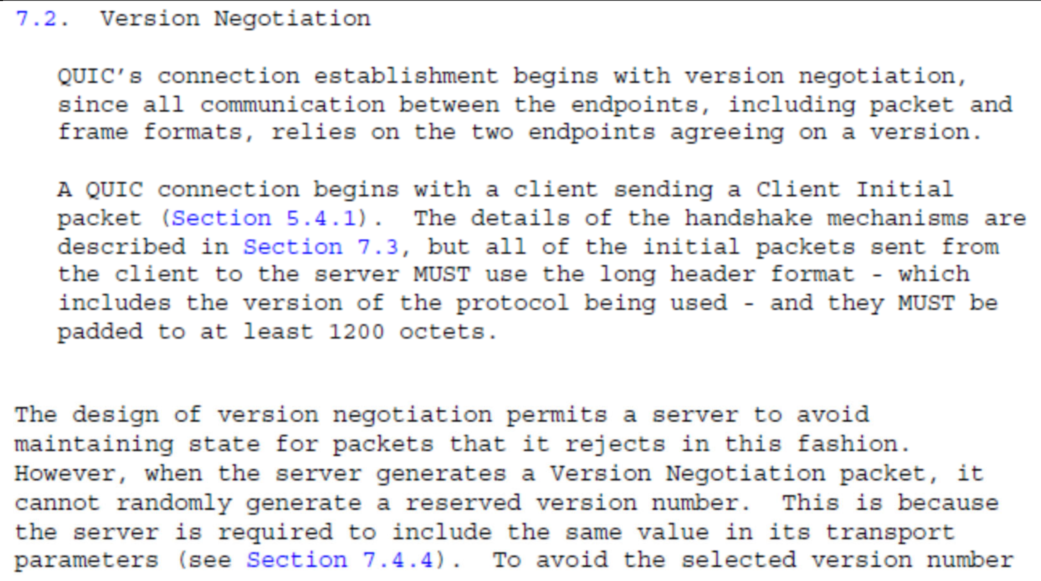
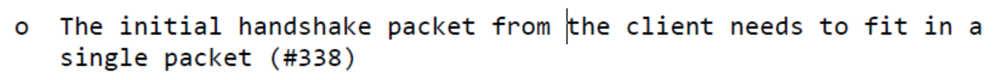
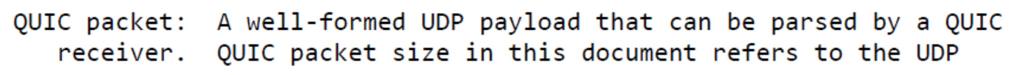
Name	St...	Protocol	Domain	Type	Initiator	Size	Ti...	Waterfall
KFOmCnqEu92Fr1...	200	http/2+quic/46	fonts.gstatic.com	font	app?origin=http...	(disk cac...	6 ...	
ListAccounts?listPa...	200	http/2+quic/46	accounts.google.com	document	rs=AA2YrTuismE...	1.4 KB	1...	
photo.jpg	200	http/2+quic/46	lh3.googleusercontent.c...	jpeg	rs=AA2YrTuismE...	(disk cac...	1...	
photo.jpg	200	http/2+quic/46	lh3.googleusercontent.c...	jpeg	rs=AA2YrTuismE...	(disk cac...	1...	
photo.jpg	200	http/2+quic/46	lh3.googleusercontent.c...	jpeg	rs=AA2YrTuismE...	(disk cac...	1...	
photo.jpg	200	http/2+quic/46	lh3.googleusercontent.c...	png	rs=AA2YrTuismE...	(disk cac...	8 ...	
m=wmwg8b	200	http/2+quic/46	www.gstatic.com	script	m=_b,_tp:434	(disk cac...	3 ...	
m=A7fCU,BVgquf,...	200	http/2+quic/46	www.gstatic.com	script	m=_b,_tp:434	(disk cac...	1...	
log?format=json&...	200	http/2+quic/46	play.google.com	xhr	m=_b,_tp:301	644 B	1...	
m=FCpbqb,OJUrv...	200	http/2+quic/46	www.gstatic.com	script	m=_b,_tp:434	(disk cac...	3 ...	
log?format=json&...	200	http/2+quic/46	play.google.com	xhr	m=_b,_tp:301	788 B	1...	
photo.jpg	200	http/2+quic/46	lh3.googleusercontent.c...	jpeg	search?safe=stric...	2.4 KB	1...	
photo.jpg	200	http/2+quic/46	lh3.googleusercontent.c...	jpeg	search?safe=stric...	2.2 KB	2...	
photo.jpg	200	http/2+quic/46	lh3.googleusercontent.c...	jpeg	search?safe=stric...	2.7 KB	1...	
photo.jpg	200	http/2+quic/46	lh3.googleusercontent.c...	png	search?safe=stric...	662 B	1...	
gen_204?atyp=i&e...	204	http/2+quic/46	www.google.com	text/html	rs=ACT90oFF1ed...	40 B	5...	
gen_204?atyp=csi...	204	http/2+quic/46	www.google.com	text/html	rs=ACT90oFF1ed...	40 B	5...	
log?format=json&...	200	http/2+quic/46	play.google.com	xhr	rs=AA2YrTuismE...	345 B	5...	

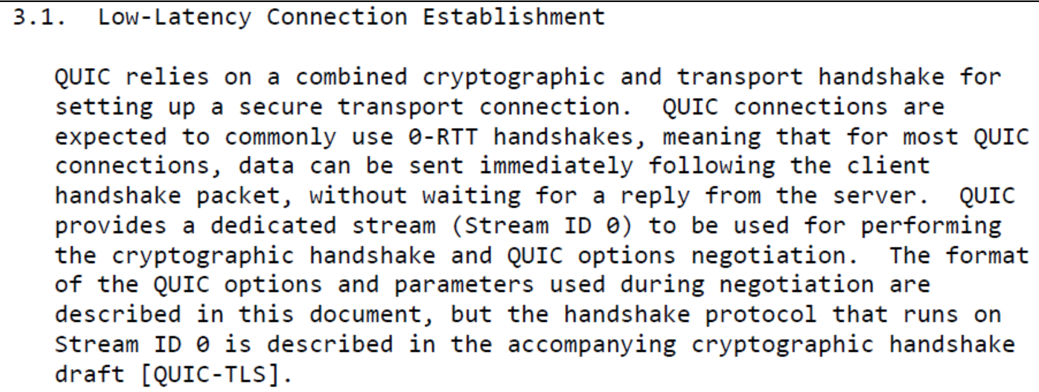
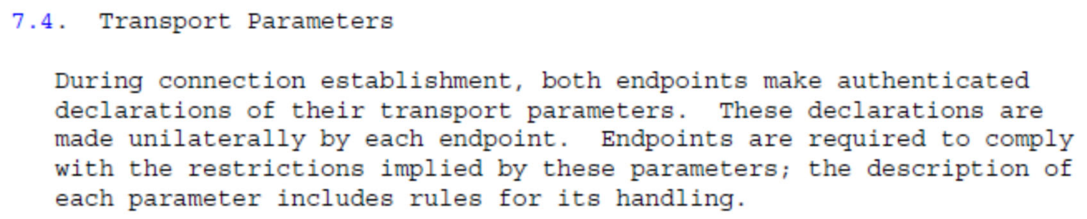
Note: As set forth below, QUIC is separate from TCP.

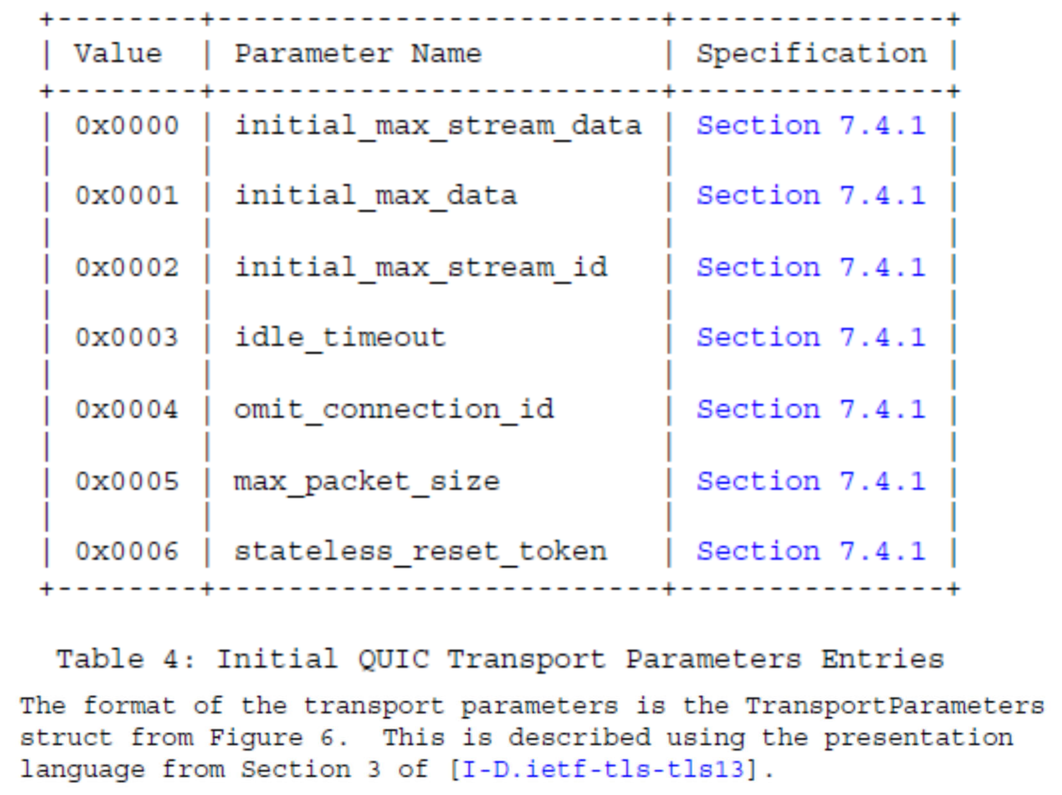

```
 1.  Introduction

   QUIC is a multiplexed and secure transport protocol that runs on top
   of UDP.  QUIC aims to provide a flexible set of features that allow
   it to be a general-purpose transport for multiple applications.

   QUIC implements techniques learned from experience with TCP, SCTP and
```

"On the surface, QUIC is very similar to TCP+TLS+HTTP/2 implemented on UDP. ...However, since QUIC is built on top of UDP, it suffers from no such limitations." https://www.chromium.org/quic

Note: As set forth below, a QUIC connection begins with a client sending a handshake packet.

	 7.2. Version Negotiation QUIC's connection establishment begins with version negotiation, since all communication between the endpoints, including packet and frame formats, relies on the two endpoints agreeing on a version. A QUIC connection begins with a client sending a Client Initial packet (Section 5.4.1). The details of the handshake mechanisms are described in Section 7.3, but all of the initial packets sent from the client to the server MUST use the long header format - which includes the version of the protocol being used - and they MUST be padded to at least 1200 octets. The design of version negotiation permits a server to avoid maintaining state for packets that it rejects in this fashion. However, when the server generates a Version Negotiation packet, it cannot randomly generate a reserved version number. This is because the server is required to include the same value in its transport parameters (see Section 7.4.4). To avoid the selected version number  o The initial handshake packet from the client needs to fit in a single packet (#338)  QUIC packet: A well-formed UDP payload that can be parsed by a QUIC receiver. QUIC packet size in this document refers to the UDP **<u>Note</u>**: As set forth below, prior to a QUIC connection being established, the QUIC connection is "set up" using the aforementioned handshake.

	 3.1. Low-Latency Connection Establishment QUIC relies on a combined cryptographic and transport handshake for setting up a secure transport connection. QUIC connections are expected to commonly use 0-RTT handshakes, meaning that for most QUIC connections, data can be sent immediately following the client handshake packet, without waiting for a reply from the server. QUIC provides a dedicated stream (Stream ID 0) to be used for performing the cryptographic handshake and QUIC options negotiation. The format of the QUIC options and parameters used during negotiation are described in this document, but the handshake protocol that runs on Stream ID 0 is described in the accompanying cryptographic handshake draft [QUIC-TLS].
detect an idle time period parameter field in the packet;	Google uses the apparatus (e.g., the one or more servers, etc.) that, when operating in accordance with the second protocol (e.g., QUIC, etc.) to establish the second protocol connection (e.g., QUIC connection), is configured to: detect an idle time period parameter field (e.g., idle timeout parameter field, etc.) in the packet (e.g., QUIC negotiation packet, etc.). See excerpt(s) below, for example (emphasis added, if any): **<u>Note</u>**: As set forth below, a QUIC negotiation packet includes transport parameters that include an idle timeout parameter that is detected by a recipient of such packet.  7.4. Transport Parameters During connection establishment, both endpoints make authenticated declarations of their transport parameters. These declarations are made unilaterally by each endpoint. Endpoints are required to comply with the restrictions implied by these parameters; the description of each parameter includes rules for its handling.

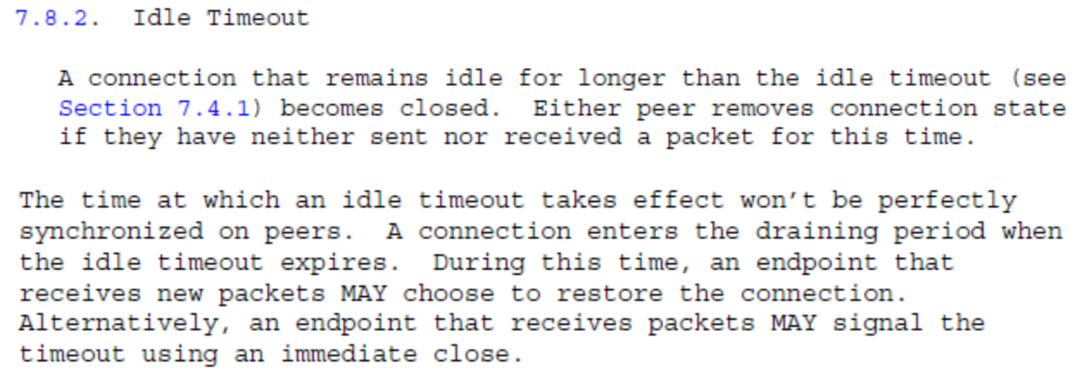
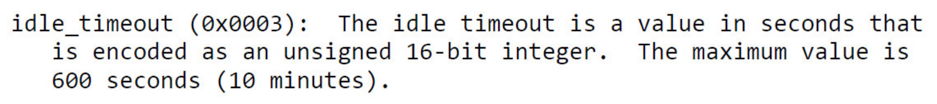
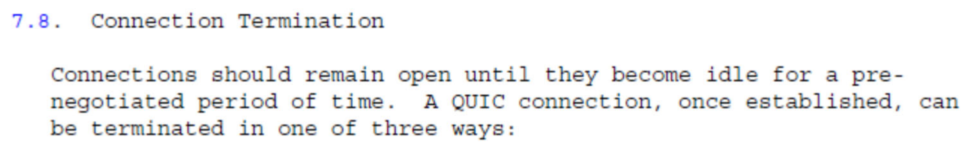
Value	Parameter Name	Specification
0x0000	initial_max_stream_data	Section 7.4.1
0x0001	initial_max_data	Section 7.4.1
0x0002	initial_max_stream_id	Section 7.4.1
0x0003	idle_timeout	Section 7.4.1
0x0004	omit_connection_id	Section 7.4.1
0x0005	max_packet_size	Section 7.4.1
0x0006	stateless_reset_token	Section 7.4.1

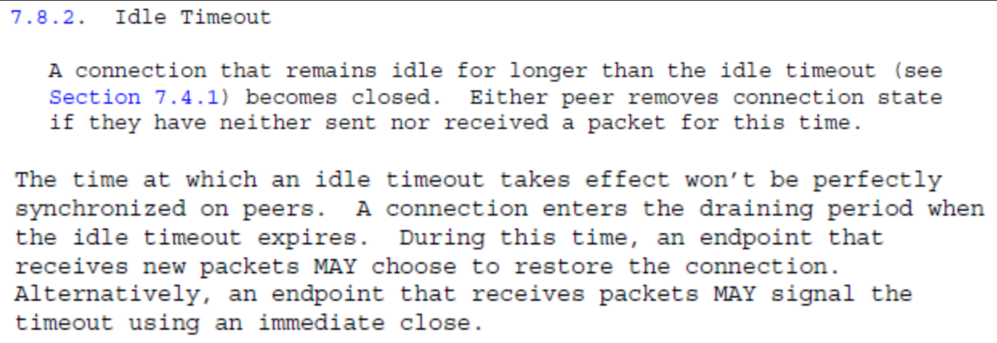
Table 4: Initial QUIC Transport Parameters Entries

The format of the transport parameters is the TransportParameters struct from Figure 6. This is described using the presentation language from Section 3 of [I-D.ietf-tls-tls13].

	```
	uint32 QuicVersion;

	enum {
	   initial_max_stream_data(0),
	   initial_max_data(1),
	   initial_max_stream_id(2),
	   idle_timeout(3),
	   omit_connection_id(4),
	   max_packet_size(5),
	   stateless_reset_token(6),
	   (65535)
	} TransportParameterId;
	```

identify metadata in the idle time period parameter field for an idle time period, detectable by the first node, where, after the idle time period is detected, the second protocol connection is deemed inactive; and	Google uses the apparatus (e.g., the one or more servers, etc.) that, when operating in accordance with the second protocol (e.g., QUIC) to establish the second protocol connection (e.g., QUIC connection), is configured to: identify metadata (e.g., a value, etc.) in the idle time period parameter field (e.g., idle timeout parameter field, etc.) for an idle time period, detectable by the first node, where, after the idle time period is detected, the second protocol connection is deemed inactive.
	
	See excerpt(s) below, for example (emphasis added, if any):
	
	Note: As set forth below, the metadata includes a value in seconds that is encoded as an unsigned 16-bit integer.
	```
	idle_timeout (0x0003):  The idle timeout is a value in seconds that
	   is encoded as an unsigned 16-bit integer.  The maximum value is
	   600 seconds (10 minutes).
	```
	Note: As set forth below, after the idle time period is detected, the QUIC connection is closed due to inactivity.

	 7.8.2. Idle Timeout A connection that remains idle for longer than the idle timeout (see Section 7.4.1) becomes closed. Either peer removes connection state if they have neither sent nor received a packet for this time. The time at which an idle timeout takes effect won't be perfectly synchronized on peers. A connection enters the draining period when the idle timeout expires. During this time, an endpoint that receives new packets MAY choose to restore the connection. Alternatively, an endpoint that receives packets MAY signal the timeout using an immediate close.
create or modify, by the second node and based on the metadata, a timeout attribute associated with the second protocol connection.	Google uses the apparatus (e.g., the one or more servers, etc.) that, when operating in accordance with the second protocol (e.g., QUIC) to establish the second protocol connection (e.g., QUIC connection), is configured to: create or modify, by the second node and based on the metadata (e.g., the value of the idle timeout parameter field, etc.), a timeout attribute associated with the second protocol connection. See excerpt(s) below, for example (emphasis added, if any): **Note**: As set forth below, since the idle_timeout value sets the duration of idleness, after which the connection is shutdown, a timeout attribute of the connect is necessarily created or modified based on the value received in the idle_timeout field of the connection negotiation packet.  idle_timeout (0x0003): The idle timeout is a value in seconds that is encoded as an unsigned 16-bit integer. The maximum value is 600 seconds (10 minutes).  7.8. Connection Termination Connections should remain open until they become idle for a pre-negotiated period of time. A QUIC connection, once established, can be terminated in one of three ways:

	 7.8.2. Idle Timeout A connection that remains idle for longer than the idle timeout (see Section 7.4.1) becomes closed. Either peer removes connection state if they have neither sent nor received a packet for this time. The time at which an idle timeout takes effect won't be perfectly synchronized on peers. A connection enters the draining period when the idle timeout expires. During this time, an endpoint that receives new packets MAY choose to restore the connection. Alternatively, an endpoint that receives packets MAY signal the timeout using an immediate close.

Caveat: The notes and/or cited excerpts utilized herein are set forth for illustrative purposes only and are not meant to be limiting in any manner. For example, the notes and/or cited excerpts, may or may not be supplemented or substituted with different excerpt(s) of the relevant reference(s), as appropriate. Further, to the extent any error(s) and/or omission(s) exist herein, all rights are reserved to correct the same in connection with any subsequent correlations.